Minor Maze Modifications

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	1024 mebibytes

A maze can be represented as a rectangle with r rows and c columns. Each cell in the maze can either contain a wall or be empty. Two cells are guaranteed to be empty: the cell at the intersection of the first row and the first column and the cell at the intersection of the r-th row and the c-th column.

It is possible to move between two empty cells if they share a side (i.e., if one coordinate is the same and the other differs by 1). It is not allowed to move into a cell with a wall.

Maria plans to remove **exactly two** walls, placing empty cells instead of them. In how many ways can she choose two distinct wall cells to be removed (order of removal does not matter) so that, after the removal, there is a path from cell $(1, 1)$ to cell (r, c) consisting only of empty cells?

Two ways to remove two walls are considered distinct if at least one of the walls is removed in one way and not removed in the other.

Input

The first line of the input contains two integers r and c ($1 \leq r, c \leq 500$). Each of the following r lines contains c characters. These lines describe the maze. The character '`X`' corresponds to a wall, and the character '`.`' corresponds to an empty cell. It is guaranteed that the first cell in the first row and the last cell in the last row are empty.

Output

Output a single integer: the number of ways to remove exactly two walls so that it is possible to move from cell $(1, 1)$ to cell (r, c).

Examples

standard input	*standard output*
3 5 ..X.. .XX.. ..XX.	7
3 2 .. X. ..	0